Computing and Software 701

Logic and Discrete Mathematics In Software Engineering

Fall 2002

Course Outline

Revised: 20 September 2002

Instructor

Dr. William M. Farmer
Office: ITB 163
Extension: 27039
E-mail: wmfarmer@mcmaster.ca
Web: http://imps.mcmaster.ca/wmfarmer
Office hours: M 9:30–10:20, M 16:30–17:20, R 9:30–10:20.

Course Web Site

http://www.cas.mcmaster.ca/~wmfarmer/CAS-701-02

Lecture Schedule

M	14:30 - 15:45	ITB 222
R	12:30 - 13:45	ITB 222

Calendar Description

"Mathematical objects and logical concepts used in Software Engineering. Higher-order logic. Partial functions and undefined terms. Practical application of the axiomatic method. Recursive definition and inductive proof. Survey of common mathematical structures and axiomatic theories used in Software Engineering. Effective use of mechanized mathematics systems."

Mission

The mission of this course is to give students a graduate-level understanding of the logic and mathematics that is needed for Software Engineering. The course will focus on the creation, use, and understanding of mathematical models. Students taking this course are expected to be familiar with propositional and first-order logic and sets, functions, and relations.

Required Text

None.

Work Plan

There will be two 75-minute lectures per week by the instructor. The lectures will present the course topics and will illustrate their use in examples. Students will be expected to attend the lectures, complete assigned exercises, and give short presentations to the class.

Grading

The exercises will not be marked. The course grade will be based on the presentations, a midterm test on October 24, and a final exam as follows:

Presentations	10%
Midterm test	30%
Final exam	60%
Total	**100%**

Policy Statements

1. Significant study and reading outside of class is required.
2. Regular class attendance is expected.
3. Students are expected to ask questions during class.
4. The midterm test may not be taken later without *prior* approval from the instructor.
5. The instructor reserves the right to require a deferred final exam to be oral.
6. Suggestions on how to improve the course and the instructor's teaching methods are always welcomed.
7. "The Faculty of Engineering is concerned with ensuring an environment that is free of all adverse discrimination. If there is a problem, that cannot be resolved by discussion among the persons concerned, individuals are reminded that they should contact their Department Chair, the Sexual Harassment/Anti-Discrimination Officer (SHADO), as soon as possible."
8. "Students are reminded that they should read and comply with the Statement on Academic Ethics and the Senate Resolutions on Academic Dishonesty as found in the Senate Policy Statements distributed at registration and available in the Senate Office" (see Senate Secretariat, Gilmour Hall, Room 104, 905-525-9140 or 905-529-7070, ext. 24337).

Syllabus

01 The Nature of Mathematics (2 hours)

- The mathematics process
- Mathematics in Software Engineering
- Mechanized mathematics systems

02 Mathematical Models (5 hours)

- Basic mathematical objects (sets, functions, and relations)
- Mathematical structures
- Axiomatic theories
- Abstract machines

03 Review of Logic (4 hours)

- Languages, models, formal systems
- Propositional logic
- First-order logic

04 Simple Type Theory (5 hours)

- Types
- Lambda notation
- Church's simple theory of types
- A Basic Extended Simple Type Theory (BESTT)
- Comparison to set theory

05 Partial Functions and Undefined Terms (3 hours)

- Techniques for formalizing partial functions
- Partial first-order logic

06 Equational Logic and Algebraic Reasoning (4 hours)

- Matching and unification
- Term rewriting
- Term models

07 Recursive Definition and Inductive Proof (5 hours)

- Primitive and well-founded recursion
- Induction principles in recursively defined structures
- Fixpoint recursion

08 Practical Application of the Axiomatic Method (4 hours)

- Lakatos' *Proofs and Refutations*
- Definitions and other conservative theory extensions
- Theory interpretation